

Oracle® Retail Furniture Retail System (FRS)
Extended Webservices Method 1 Guide
Release 1101

October 2015

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Primary Author: John Hardwick

Contributors:

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the **MicroStrategy** Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(ii) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(iii) the software component known as **Access Via**TM licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(iv) the software component known as **Adobe Flex**TM licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

Send Us Your Comments	vii
Preface	ix
Audience.....	ix
Documentation Accessibility.....	ix
Related Documents.....	ix
Customer Support.....	ix
Improved Process for Oracle Retail Documentation Corrections.....	ix
Oracle Retail Documentation on the Oracle Technology Network.....	x
Conventions.....	x
Extended Webservices Methods 1 Guide	1
Overview.....	1
Web Methods.....	1
GetBaseProducts.....	1
GetOptionQuestions.....	4
GetNestedOptionQuestions.....	7
AddOrderLineForBaseProduct.....	10

Send Us Your Comments

Oracle Retail Furniture Retail System (FRS), Extended Webservices Method 1 Guide,
Release 1101

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Technology Network Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Audience

This user guide is for users and administrators of Oracle Retail Furniture Retail System (FRS). This includes merchandisers, buyers, business analysts, and administrative personnel.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Retail Furniture Retail System (FRS) Release 1101 documentation set:

- Oracle Retail Furniture Retail System (FRS) Installation Guide
- Oracle Retail Furniture Retail System (FRS) Release Notes
- Oracle Retail Furniture Retail System (FRS) Operations Guide
- Oracle Retail Furniture Retail System (FRS) Online Help

Customer Support

- To contact Oracle Customer Support, access My Oracle Support at the following URL:
- <https://support.oracle.com>
- When contacting Customer Support, please provide the following:
- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times **not** be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the

case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

<http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html>

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-02 is an updated version of a document with part number E123456-01.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Technology Network

Oracle Retail product documentation is available on the following web site:

<http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html>

(Data Model documents are not available through Oracle Technology Network. You can obtain them through My Oracle Support.)

Conventions

Navigate: This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement “the Window Name window opens.”

This is a code sample

It is used to display examples of code

Extended Webservices Methods 1 Guide

Overview

This document details web service integration methods

Web Methods

GetBaseProducts

The GetBaseProducts web method is an overloaded method that is used to get either an array of BaseProduct objects that represents the entire set of web shop base products, or an array of BaseProduct objects that represents a partial set of web shop base products whose "SKU" matches the string of text passed to the method. It should be noted that in a .Net Framework development environment the implementation of an overloaded web method is seamless. That is, the proxy service exposes the methods as GetProducts() and GetProducts(string) which is what was intended. However, the SOAP specification requires each method to have a distinct name. Therefore there are two SOAP definitions for the one method; GetProducts and GetProductsForSKUSearchString

Parameters

SKUSearchString [IN] String	A string representing the SKU to be searched. SQL wild card characters "%" and "_" may be used. This parameter is optional.
return [OUT] ArrayOfProduct	A successful call returns an array of BaseProduct objects whose SKU matches SKUSearchString. If there are no products that match SKUSearchString then nothing is returned. If SKUSearchString is not passed the an array of all webshop base products is returned

Error Codes

There are no error codes associated with this method.

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /frsnetwebservices/webshoppervice.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
"http://www.radii.co.uk/FRSNetWebServices/WebShopService/GetBaseProducts"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetBaseProducts
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService" />
  </soap:Body>
</soap:Envelope>
```

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetBaseProductsResponse
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <GetBaseProductsResult>
        <BaseProduct>
          <BaseProductID>int</BaseProductID>
          <SKU>string</SKU>
          <Supplier>string</Supplier>
          <Range>string</Range>
          <ModelNumber>string</ModelNumber>
          <Description1>string</Description1>
          <Description2>string</Description2>
          <Description3>string</Description3>
          <Width>string</Width>
          <Depth>string</Depth>
          <Height>string</Height>
          <Weight>string</Weight>
          <BaseSellingPrice>decimal</BaseSellingPrice>
          <VATID>int</VATID>
          <VATRate>decimal</VATRate>
          <LeadTime>int</LeadTime>
          <PromotionID>int</PromotionID>
          <ImageURL>string</ImageURL>
          <HasOptions>boolean</HasOptions>
          <ProductGroupCode>string</ProductGroupCode>
          <ProductGroupDescription>string</ProductGroupDescription>
          <Barcode>string</Barcode>
          <Feature>string</Feature>
          <SortNumber>int</SortNumber>
        </BaseProduct>
        <BaseProduct>
          <BaseProductID>int</BaseProductID>
          <SKU>string</SKU>
          <Supplier>string</Supplier>
          <Range>string</Range>
          <ModelNumber>string</ModelNumber>
          <Description1>string</Description1>
          <Description2>string</Description2>
          <Description3>string</Description3>
          <Width>string</Width>
          <Depth>string</Depth>
          <Height>string</Height>
          <Weight>string</Weight>
          <BaseSellingPrice>decimal</BaseSellingPrice>
          <VATID>int</VATID>
          <VATRate>decimal</VATRate>
          <LeadTime>int</LeadTime>
          <PromotionID>int</PromotionID>
          <ImageURL>string</ImageURL>
          <HasOptions>boolean</HasOptions>
          <ProductGroupCode>string</ProductGroupCode>
          <ProductGroupDescription>string</ProductGroupDescription>
          <Barcode>string</Barcode>
          <Feature>string</Feature>
          <SortNumber>int</SortNumber>
        </BaseProduct>
      </GetBaseProductsResult>
    </GetBaseProductsResponse>
  </soap:Body>
</soap:Envelope>

```

```

        </BaseProduct>
    </GetBaseProductsResult>
</GetBaseProductsResponse>
</soap:Body>
</soap:Envelope>

```

Sample Code

The following code demonstrates how to get a base product with an SKU of "9-1-153".

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        private static FRSNetWebServices.WebShopService ws;

        [STAThread]
        static void Main(string[] args)
        {
            ws = new FRSNetWebServices.WebShopService();
            //always initialise a CookieContainer instance to hold the session
            ws.CookieContainer = new System.Net.CookieContainer();

            string sku = "9-1-153";

            FRSNetWebServices.BaseProduct[] products = ws.GetBaseProducts(sku);

            foreach (FRSNetWebServices.BaseProduct p in products)
            {
                Console.WriteLine("BaseProductID: {0}",
p.BaseProductID.ToString());
                Console.WriteLine("SKU: {0}", p.SKU);
                Console.WriteLine("Supplier: {0}", p.Supplier);
                Console.WriteLine("Range: {0}", p.Range);
                Console.WriteLine("ModelNumber: {0}", p.ModelNumber);
                Console.WriteLine("Description1: {0}", p.Description1);
                Console.WriteLine("Description2: {0}", p.Description2);
                Console.WriteLine("Description3: {0}", p.Description3);
                Console.WriteLine("Width: {0}", p.Width);
                Console.WriteLine("Depth: {0}", p.Depth);
                Console.WriteLine("Height: {0}", p.Height);
                Console.WriteLine("Weight: {0}", p.Weight);
                Console.WriteLine("BaseSellingPrice: {0}",
p.BaseSellingPrice.ToString());
                Console.WriteLine("VATID: {0}", p.VATID.ToString());
                Console.WriteLine("VATRate: {0}", p.VATRate.ToString());
                Console.WriteLine("LeadTime: {0}", p.LeadTime.ToString());
                Console.WriteLine("PromotionID: {0}", p.PromotionID.ToString());
                Console.WriteLine("ImageURL: {0}", p.ImageURL);
                Console.WriteLine("HasOptions: {0}", p.HasOptions.ToString());
                Console.WriteLine("ProductGroupCode: {0}", p.ProductGroupCode);
                Console.WriteLine("ProductGroupDescription: {0}",
p.ProductGroupDescription);
                Console.WriteLine("Barcode: {0}", p.Barcode);
                Console.WriteLine("Feature: {0}", p.Feature);
                Console.WriteLine("SortNumber: {0}", p.SortNumber.ToString());
            }
        }
    }
}

```

```
}
}
}
OUTPUT:
BaseProductID: 241121
SKU: 9-1-153
Supplier: Chipton Furniture'"
Range: Mellow
ModelNumber: 273
Description1: 3' Canted
Description2: Display Unit
Description3:
Width:
Depth:
Height:
Weight:
BaseSellingPrice: 849.0000
VATID: 2
VATRate: 0.175000
LeadTime: 42
PromotionID: 554
ImageURL: mcr13.jpg
HasOptions: True
ProductGroupCode: D
ProductGroupDescription: Dining
Barcode:
Feature:
SortNumber: 5
```

GetOptionQuestions

The GetOptionQuestions method is used to return a collection of Option Questions associated with a base product.

Parameters

BaseProductID [IN] Int	An integer representing the ProductID of the base product.
PromotionID [IN] Int	An integer representing the PromotionID of the active promotion for the base product.
return [OUT] ArrayOfOptionQuestions	A successful call to GetCustomer returns a Customer object representing the customer whose CustomerID was passed to the method. If CustomerID is not found then nothing is returned

Error Codes

There are no Error codes associated with this method.
The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /FRSNetWebServices/WebShopService.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
```

```

"http://www.radii.co.uk/FRSNetWebServices/WebShopService/GetOptionQuestions"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetOptionQuestions
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <BaseProductID>int</BaseProductID>
      <PromotionID>int</PromotionID>
    </GetOptionQuestions>
  </soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetOptionQuestionsResponse
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <GetOptionQuestionsResult>
        <OptionQuestion>
          <OptionQuestionID>int</OptionQuestionID>
          <Descriptor>string</Descriptor>
          <OptionType>NotSet or Choice or FreeText</OptionType>
          <OptionQuantityType>NotSet or OneOnly or ZeroToTen or
AnyValue</OptionQuantityType>
          <OptionPriceType>NotSet or MonetaryValue or
PercentageIncrease</OptionPriceType>
          <OptionChoices>
            <OptionChoice xsi:nil="true" />
            <OptionChoice xsi:nil="true" />
          </OptionChoices>
        </OptionQuestion>
        <OptionQuestion>
          <OptionQuestionID>int</OptionQuestionID>
          <Descriptor>string</Descriptor>
          <OptionType>NotSet or Choice or FreeText</OptionType>
          <OptionQuantityType>NotSet or OneOnly or ZeroToTen or
AnyValue</OptionQuantityType>
          <OptionPriceType>NotSet or MonetaryValue or
PercentageIncrease</OptionPriceType>
          <OptionChoices>
            <OptionChoice xsi:nil="true" />
            <OptionChoice xsi:nil="true" />
          </OptionChoices>
        </OptionQuestion>
      </GetOptionQuestionsResult>
    </GetOptionQuestionsResponse>
  </soap:Body>
</soap:Envelope>

```

Sample Code

The following code demonstrates how to get the first level of option questions for a base product.

```
namespace ConsoleApplication1
```

```

{
    using FRSNetWebServices;
    class Class1
    {
        private static FRSNetWebServices.WebShopService ws;

        [STAThread]
        static void Main(string[] args)
        {
            ws = new FRSNetWebServices.WebShopService();
            //always initialise a CookieContainer instance to hold the session
            ws.CookieContainer = new System.Net.CookieContainer();

            string sku = "9-1-153";

            FRSNetWebServices.BaseProduct[] products =
ws.GetBaseProducts(sku);

            FRSNetWebServices.BaseProduct p = products[0];

            FRSNetWebServices.OptionQuestion[] optionQuestions =
ws.GetOptionQuestions(p.BaseProductID, p.PromotionID);

            foreach (FRSNetWebServices.OptionQuestion o in optionQuestions)
            {
                Console.WriteLine("OptionQuestionID: {0}",
o.OptionQuestionID.ToString());
                Console.WriteLine("Descriptor: {0}", o.Descriptor);
                Console.WriteLine("OptionType: {0}", o.OptionType);
                Console.WriteLine("OptionQuantityType: {0}",
o.OptionQuantityType);
                Console.WriteLine("OptionPriceType: {0}",
o.OptionPriceType);

                foreach (FRSNetWebServices.OptionChoice c in
o.OptionChoices)
                {
                    Console.WriteLine("OPTION CHOICE");
                    Console.WriteLine("OptionID: {0}",
c.OptionID.ToString());
                    Console.WriteLine("Value: {0}", c.Value);
                    Console.WriteLine("SellingPrice: {0}",
c.SellingPrice.ToString());
                    Console.WriteLine("HasOptions: {0}",
c.HasOptions.ToString());
                }
            }
        }
    }
}

```

OUTPUT:

```

OptionQuestionID: 3
Descriptor: Finish.
OptionType: Choice
OptionQuantityType: OneOnly
OptionPriceType: MonetaryValue
OPTION CHOICE
OptionID: 1304
Value: Mahogany
SellingPrice: 25.00
HasOptions: True

```



```
OPTION CHOICE
OptionID: 1305
Value: Cherry
SellingPrice: 125.00
HasOptions: True
OPTION CHOICE
OptionID: 1306
Value: Yew
SellingPrice: 0
HasOptions: False
```

GetNestedOptionQuestions

The GetNestedOptionQuestions method is used to return a collection of Option Questions associated with an Option Choice.

Parameters

BaseProductID [IN] Int	An integer representing the ProductID of the base product.
OptionID [IN] Int	An integer representing the OptionID of the Option Choice.
PromotionID [IN] Int	An integer representing the PromotionID of the active promotion for the base product.
return [OUT] ArrayOfOptionQuestions	A successful call to GetNestedOptionQuestions returns an array of OptionQuestion objects.

Error Codes

There are no Error codes associated with this method.

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /frsnetwebservices/webshoppervice.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
"http://www.radii.co.uk/FRSNetWebServices/WebShopService/GetNestedOptionQuestions"
"
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNestedOptionQuestions
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <BaseProductID>int</BaseProductID>
      <OptionID>int</OptionID>
      <PromotionID>int</PromotionID>
    </GetNestedOptionQuestions>
  </soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
```

```

Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetNestedOptionQuestionsResponse
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <GetNestedOptionQuestionsResult>
        <OptionQuestion>
          <OptionQuestionID>int</OptionQuestionID>
          <Descriptor>string</Descriptor>
          <OptionType>NotSet or Choice or FreeText</OptionType>
          <OptionQuantityType>NotSet or OneOnly or ZeroToTen or
AnyValue</OptionQuantityType>
          <OptionPriceType>NotSet or MonetaryValue or
PercentageIncrease</OptionPriceType>
          <OptionChoices>
            <OptionChoice xsi:nil="true" />
            <OptionChoice xsi:nil="true" />
          </OptionChoices>
        </OptionQuestion>
        <OptionQuestion>
          <OptionQuestionID>int</OptionQuestionID>
          <Descriptor>string</Descriptor>
          <OptionType>NotSet or Choice or FreeText</OptionType>
          <OptionQuantityType>NotSet or OneOnly or ZeroToTen or
AnyValue</OptionQuantityType>
          <OptionPriceType>NotSet or MonetaryValue or
PercentageIncrease</OptionPriceType>
          <OptionChoices>
            <OptionChoice xsi:nil="true" />
            <OptionChoice xsi:nil="true" />
          </OptionChoices>
        </OptionQuestion>
      </GetNestedOptionQuestionsResult>
    </GetNestedOptionQuestionsResponse>
  </soap:Body>
</soap:Envelope>

```

Sample Code

The following code demonstrates how to get the first level of option questions and choices for a base product and then get the nested options for each choice if the choice has nested options.

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        private static FRSNetWebServices.WebShopService ws;

        [STAThread]
        static void Main(string[] args)
        {
            ws = new FRSNetWebServices.WebShopService();
            //always initialise a CookieContainer instance to hold the session

```

```

ws.CookieContainer = new System.Net.CookieContainer();

string sku = "9-1-153";

FRSNetWebServices.BaseProduct[] products = ws.GetBaseProducts(sku);

FRSNetWebServices.BaseProduct p = products[0];

GetAllOptions(p);

}

static void GetAllOptions(FRSNetWebServices.BaseProduct p)
{
    //get first level options
    FRSNetWebServices.OptionQuestion[] optionQuestions =
ws.GetOptionQuestions(p.BaseProductID, p.PromotionID);

    WriteOptionsAndChoices(optionQuestions, p);
}

static void WriteOptionsAndChoices(FRSNetWebServices.OptionQuestion[]
optionQuestions, FRSNetWebServices.BaseProduct p)
{
    foreach (FRSNetWebServices.OptionQuestion o in optionQuestions)
    {
        Console.WriteLine("OptionQuestionID: {0}",
o.OptionQuestionID.ToString());
        Console.WriteLine("Descriptor: {0}", o.Descriptor);
        Console.WriteLine("OptionType: {0}", o.OptionType);
        Console.WriteLine("OptionQuantityType: {0}",
o.OptionQuantityType);
        Console.WriteLine("OptionPriceType: {0}", o.OptionPriceType);

        foreach (FRSNetWebServices.OptionChoice c in o.OptionChoices)
        {
            Console.WriteLine("OPTION CHOICE");
            Console.WriteLine("OptionID: {0}", c.OptionID.ToString());
            Console.WriteLine("Value: {0}", c.Value);
            Console.WriteLine("SellingPrice: {0}",
c.SellingPrice.ToString());
            Console.WriteLine("HasOptions: {0}",
c.HasOptions.ToString());

            if (c.HasOptions)
            {
                FRSNetWebServices.OptionQuestion[] nestedOptionQuestions
= ws.GetNestedOptionQuestions(p.BaseProductID, c.OptionID, p.PromotionID);
                Console.WriteLine("");
                Console.WriteLine("NESTED OPTIONS");
                WriteOptionsAndChoices(nestedOptionQuestions, p);
            }
        }
    }
}
}
}
}
}

```

AddOrderLineForBaseProduct

The AddOrderLine web method has an overloaded signature that implements the SOAP method AddOrderLineForBaseProduct and is used to add a base product with option answers to the current session's new sales order lines array which is persisted by the web service. The AddOrderLineForBaseProduct method requires information about the selected product. The parameter values should be set to the corresponding Product property value where applicable. For example, if ProductID is set to Product.ProductID then VATID should be set to Product.VATID. See the parameter table below which shows from where the values should originate. Note that nothing is returned by this method. If the method completes without error then the method has completed successfully.

Parameters

ProductID [IN] int	Required. Must be a valid BaseProduct.ProductID
ArrayOfOptionAnswers [IN] string	An array of option answers. Note that an OptionAnswer can also have an array of option answers as a property if the option choice itself has further option questions.
Quantity [IN] decimal	Required
BaseSellingPrice [IN] decimal	Required. (BaseProduct.BaseSellingPrice)
VATID [IN] int	Required. (BaseProduct.VATID)
TakeFromStock [IN] Boolean	If set to True then indicates that free stock should be reserved immediately when the order is inserted provided there is enough free stock to fulfil the entire order line quantity.
LeadTime [IN] int	Product.Leadtime
PromotionID [IN] int	Product.PromotionID

SOAP

The following is a sample SOAP request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /frsnetwebservices/webshopservice.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
"http://www.radii.co.uk/FRSNetWebServices/WebShopService/AddOrderLineForBaseProduct"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddOrderLineForBaseProduct
xmlns="http://www.radii.co.uk/FRSNetWebServices/WebShopService">
      <BaseProductID>int</BaseProductID>
      <optionAnswers>
        <OptionAnswer>
          <OptionQuestionID>int</OptionQuestionID>
```

```

        <Quantity>decimal</Quantity>
        <SellingPrice>decimal</SellingPrice>
    </OptionAnswer>
    <OptionAnswer>
        <OptionQuestionID>int</OptionQuestionID>
        <Quantity>decimal</Quantity>
        <SellingPrice>decimal</SellingPrice>
    </OptionAnswer>
</optionAnswers>
<Quantity>int</Quantity>
<BaseSellingPrice>decimal</BaseSellingPrice>
<VATID>int</VATID>
<TakeFromStock>boolean</TakeFromStock>
<LeadTime>int</LeadTime>
<PromotionID>int</PromotionID>
</AddOrderLineForBaseProduct>
</soap:Body>
</soap:Envelope>

```

Sample Code

The following code demonstrates how to get a base product with selectable options and get the product's option questions and nested option questions. An array of option answers is constructed and the base product is added to the session's list of sales order lines using the overloaded AddOrderLine method. The order price is calculated and a payment taken. The sales order is then saved to the database by calling the InsertSalesOrder method.

```

using System;
using System.Collections.Generic;
using System.Text;
using System.Web.Services.Protocols;

namespace ConsoleApplication1
{
    class Program
    {
        private static FRSNetWebServices.WebShopService ws;

        [STAThread]
        static void Main(string[] args)
        {
            ws = new FRSNetWebServices.WebShopService();
            //always initialise a CookieContainer instance to hold the session
            ws.CookieContainer = new System.Net.CookieContainer();

            string sku = "9-1-153";
            int customerID = 1389;
            decimal totalOptionPrice = 0;
            decimal totalPrice;

            FRSNetWebServices.BaseProduct[] products = ws.GetBaseProducts(sku);
            FRSNetWebServices.BaseProduct p = products[0];

            try
            {
                ws.AddOrderLine(p.BaseProductID, ChooseOptions(p, ref
                totalOptionPrice), 1, p.BaseSellingPrice,p.VATID,true,p.LeadTime,p.PromotionID);
            }
            catch (SoapException myExc)
            {
            }
        }
    }
}

```

```

        Console.WriteLine("SOAP EXCEPTION. DETAIL " +
myExc.Detail.OuterXml + " " + myExc.Detail.InnerXml);
    }

    totalPrice = p.BaseSellingPrice + totalOptionPrice;

    try
    {
        ws.AddPayment(4,totalPrice,"*****6789");
    }
    catch (SoapException myExc)
    {
        Console.WriteLine("SOAP EXCEPTION. DETAIL " +
myExc.Detail.OuterXml + " " + myExc.Detail.InnerXml);
    }

    try
    {
        ws.InsertSalesOrder(customerID, "ORDER REFERENCE",
DateTime.Today.AddDays(p.LeadTime), false, true, 2, false);
    }
    catch (SoapException myExc)
    {
        Console.WriteLine("SOAP EXCEPTION. DETAIL " +
myExc.Detail.OuterXml + " " + myExc.Detail.InnerXml);
    }
}

static FRSNetWebServices.OptionAnswer[]
ChooseOptions(FRSNetWebServices.BaseProduct p, ref decimal totalOptionPrice)
{
    //get first level options
    FRSNetWebServices.OptionQuestion[] optionQuestions =
ws.GetOptionQuestions(p.BaseProductID, p.PromotionID);

    return GetAnswers(optionQuestions, p, ref totalOptionPrice);
}

static FRSNetWebServices.OptionAnswer[]
GetAnswers(FRSNetWebServices.OptionQuestion[] optionQuestions,
FRSNetWebServices.BaseProduct p, ref decimal totalOptionPrice)
{
    FRSNetWebServices.OptionAnswer[] answers = new
FRSNetWebServices.OptionAnswer[optionQuestions.GetLength(0)];

    int index = 0;
    foreach (FRSNetWebServices.OptionQuestion o in optionQuestions)
    {
        if (o.OptionType == FRSNetWebServices.OptionQuestionType.Choice)
        {
            //choose the first option
            FRSNetWebServices.OptionChoice c = o.OptionChoices[0];
            {
                FRSNetWebServices.OptionAnswerForChoice ansChoice = new
FRSNetWebServices.OptionAnswerForChoice();

                ansChoice.OptionQuestionID = o.OptionQuestionID;
                ansChoice.OptionID = c.OptionID;
            }
        }
    }
}

```

```

        switch (o.OptionQuantityType)
        {
            case
ConsoleApplication1.FRSNetWebServices.OptionQuestionQuantityType.OneOnly:
                ansChoice.Quantity = 1;
                break;
            case
ConsoleApplication1.FRSNetWebServices.OptionQuestionQuantityType.ZeroToTen:
                ansChoice.Quantity = 10;
                break;
            case
ConsoleApplication1.FRSNetWebServices.OptionQuestionQuantityType.AnyValue:
                ansChoice.Quantity = 99;
                break;
        }

        switch (o.OptionPriceType)
        {
            case
ConsoleApplication1.FRSNetWebServices.OptionQuestionPriceType.MonetaryValue:
                ansChoice.SellingPrice = c.SellingPrice;
                break;
            case
ConsoleApplication1.FRSNetWebServices.OptionQuestionPriceType.PercentageIncrease:
                ansChoice.SellingPrice = c.SellingPrice *
p.BaseSellingPrice / 100;
                break;
        }

        if (c.HasOptions)
        {
            ansChoice.OptionAnswers =
GetAnswers(ws.GetNestedOptionQuestions(p.BaseProductID, c.OptionID,
p.PromotionID), p, ref totalOptionPrice);
        }

        answers[index] = ansChoice;
    }

    }
    if (o.OptionType ==
FRSNetWebServices.OptionQuestionType.FreeText)
    {
        FRSNetWebServices.OptionAnswerForFreeText ansText = new
FRSNetWebServices.OptionAnswerForFreeText();
        ansText.OptionQuestionID = o.OptionQuestionID;
        ansText.FreeText = "Text input by user";
        ansText.SellingPrice = 0;
        ansText.Quantity = 1;

        answers[index] = ansText;
    }

    totalOptionPrice += answers[index].SellingPrice;
    index += 1;
}

return answers;
}
}
}

```

